CERTIFICATE OF TRANSMISSION

I hereby certify that this correspondence (along with any paper referred to as being attached or enclosed) is being submitted *via* the USPTO EFS Filing System on the date shown below to **Mail Stop Appeal Brief - Patents**, Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Date: December 18, 2008 /Jessica Sexton/
Jessica Sexton

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re patent application of:			
Applicant(s):	Steven Bailey, *et al.*	Examiner:	Fred I. Ehichioya
Serial No:	10/826,517	Art Unit:	2162
Filing Date:	April 16, 2004		

Title: SYSTEM AND METHODS FOR DATABASE LOCK WITH REFERENCE COUNTING

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Dear Sir:

Applicant submits this brief in connection with an appeal of the above-identified patent application. Payment is being submitted via credit card in connection with all fees due regarding this appeal brief. In the event any additional fees may be due and/or are not covered by the credit card, the Commissioner is authorized to charge such fees to Deposit Account No. 50-1063 [MSFTP622US].

I. Real Party in Interest (37 C.F.R. §41.37(c)(1)(i))

The real party in interest in the present appeal is Microsoft Corporation, the assignee of the present application.

II. Related Appeals and Interferences (37 C.F.R. §41.37(c)(1)(ii))

Appellants, appellants' legal representative, and/or the assignee of the present application are not aware of any appeals or interferences which may be related to, will directly affect, or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. Status of Claims (37 C.F.R. §41.37(c)(1)(iii))

Claims 1-3, 5-18, 20, 21, 23, 24 and 26-30 stand rejected by the Examiner. Claims 4, 19, 22, and 25 have been cancelled while claim 31 is withdrawn. The rejection of claims 1-3, 5-18, 20, 21, 23, 24 and 26-30 is being appealed.

IV. Status of Amendments (37 C.F.R. §41.37(c)(1)(iv))

The Examiner has not entered the amendments submitted after the Final Office Action. (*See* Communication from Examiner dated July 2, 2008).

V. Summary of Claimed Subject Matter (37 C.F.R. §41.37(c)(1)(v))

A. Independent claim 1

Independent claim 1 recites a computer implemented system for managing the access of system resources in a database comprising the following computer executable components: a lock manager that acquires a parent lock and one or more child locks on resource(s) of a database (*See*, Fig 1 and text at pg. 7 line 25 to pg. 8 line 37), the lock manager stores a reference count of the one or more child locks within the parent lock such that as each child lock is released, the reference count decrements by a value of one, and the parent lock is released upon release of all child locks associated therewith (*See*, Fig 4 and text at pg. 11 lines 12 to 23).

B. Independent claim 12

Independent claim 12 recites a computer implemented method for controlling locks to manage access to system resources in a database comprising: defining a parent-child relation ship among a plurality of locks in a lock hierarchy; reference counting one or more child locks associated with a parent lock, such that a parent lock maintains a count reference of respective child locks associated therewith (*See*, Fig 1 and text at pg. 7 line 25 to pg. 8 line 37) and as each child lock is released, the reference count decrements by a value of one; and, releasing a parent lock upon a release of all the respective child locks associated therewith (*See*, Fig 4 and text at pg. 11 lines 12 to 23).

C. Independent claim 20

Independent claim 20 recites a computer implemented database management system comprising: locking means for locking a resource on a database; means for counting one or more child locks associated with the locking means, wherein the counting means is decreased by one as each child lock is released (*See*, Fig 4 and text at pg. 11 lines 12 to 23); and means for determining a lifetime of the locking means based on the number of child locks associated therewith (*See*, Figs 6, 7 and text at pg. 13 lines 4 to 31).

The aforementioned means for limitations are identified as claim elements subject to the provisions of 35 U.S.C. §112 ¶6. The corresponding structures are identified with reference to the specification and drawings in the parentheticals above corresponding to those claim limitations.

D. Independent claim 21

Independent claim 21 recites a computer implemented method for controlling locks to manage access to system resources in a database comprising: counting one or more child locks associated with a parent lock to obtain a reference count of the child locks associated therewith; releasing a child lock; decrementing the reference count by a value of one; and releasing the parent lock upon the reference count reaching a zero value (*See*, Fig 4 and text at pg. 11 lines 12 to 23).

E. Independent claim 26

Independent claim 26 recites a computer implemented database lock management system for managing access to system resources comprising: a computer executable lock manager that acquires at least a parent lock and one or more child locks on a database resource, the lock manager creates within the parent lock a reference count of the child lock so that (*See*, Fig 1 and text at pg. 7 line 25 to pg. 8 line 37) the lock manager releases the parent lock upon the reference count attainment of a zero value (*See*, Fig 4 and text at pg. 11 lines 12 to 23).

VI. Grounds of Rejection to be Reviewed (37 C.F.R. §41.37(c)(1)(vi))

A. Whether claims 1-11 and 26-30 are unpatentable under 35 U.S.C. §101 as directed to non-statutory subject matter.

B. Whether claims 1, 5, 8, 9, 11-14, 17, 21 and 23-24 are unpatentable under 35 U.S.C. §103(a) over Ashok M. Joshi (US 5,414,839) in view of Non-Patent Literature "Concurrent Access to Point Data" by Ng, *et al.*

C. Whether claims 2, 3, 10, 15, 18 and 20 are unpatentable under 35 U.S.C. §103(a) over Joshi in view of Ng, *et al.* and further in view of Chan, *et al.* (US 6,108,654).

D. Whether claims 6 and 16 are unpatentable under 35 U.S.C. §103(a) over Joshi in view of Ng, *et al.* and further in view of Bray, *et al.*

E. Whether claims 26-30 are unpatentable under 35 U.S.C. §103(a) over Joshi in view of Chan, *et al.* (US 6,108,654).

VII. Argument (37 C.F.R. §41.37(c)(1)(vii))

A. Rejection of Claims 1-11 and 26-30 Under 35 U.S.C. §101

Claims 1-11 and 26-30 stand rejected under 35 U.S.C. §101 because the claimed invention is directed to non-statutory subject matter. Reversal of this rejection is respectfully

requested for at least the following reasons. Claims 1 and 26 recite a computer implemented systems and methods that are utilized in management of locks on database resources. The claims pertain to a computer executable lock manager that stores a reference count of child locks within a parent locks so that the parent lock is released upon releasing all the child locks. Such a system produces a concrete, tangible result of a reference count of child locks stored within parent locks, which is useful for determining the lifetime of the parent locks. Accordingly, it is submitted that the computer implemented system and method recited in the subject claims produces a useful, concrete and tangible result and is therefore statutory.

B. Rejection of Claims 1, 5, 8, 9, 11-14, 17, 21 and 23-24 Under 35 U.S.C. §103(a)

Claims 1, 5, 8, 9, 11-14, 17, 21 and 23-24 stand rejected under 35 U.S.C. §103(a) over Ashok M. Joshi (US 5,414,839) in view of Non-Patent Literature "Concurrent Access to Point Data" by Ng, *et al.* Reversal of this rejection is requested for at least the following reasons. Joshi or Ng *et al.* alone or in combination fail to teach or suggest each and every feature of the subject invention.

Appellants' claimed subject matter relates to systems and methods for supplying a database with a parent-child lock hierarchy arrangement, such that each lock contains sufficient information to determine its own lifetime. In particular, amended independent claim 1 recites *a computer implemented database management system comprising a lock manager that acquires a parent lock and one or more child locks on resource(s) of a database, the lock manager stores a reference count of the one or more child locks within the parent lock such that,* ***as each child lock is released, the reference count decrements by a value of one and the parent lock is released upon release of all child locks associated therewith.*** Independent claims 12 and 21 recite similar features. Joshi and Ng *et al.* do not disclose or suggest such features recited by the subject claims.

Joshi relates to a database system that uses lock escalation and de-escalation protocols in a concurrency control mechanism. A lock granularity tree with a root node and leaf nodes of lower granularity, organizes records of a table into a hierarchy. At the cited portion on col.12, lines 56-57, Joshi discloses the child lock being released, but is silent regarding the step taken *as each child lock is released.* At page 5 of the Final Office Action, the Examiner concedes that

Joshi does not teach the reference count decrements by a value of one and the parent lock is released upon the release of all child locks associated therewith. The Examiner cites Ng *et al.* to cure the aforementioned deficiencies of Joshi.

Ng *et al.* relates to a lock-coupling technique to B+-tree, the R-tree and the KDB tree to allow concurrent operations. At the cited portions, Ng *et al.* discloses a method for avoiding any change to a parent node before the locking of its child nodes is completed. For every parent node a p lock on the node must be acquired and held until the p locks on the child nodes to be visited are acquired (*See.* pg. 370, section 4.1 first paragraph). A counter in the parent is initialized to the number of subsearch operations initiated at a node, and is decremented whenever a child lock is p-locked (*See.* pg. 369, section 3.1 second paragraph). When the counter value is zero, *i.e.* all the child nodes are p-locked, the lock on the parent is released. Further at the cited portions of item 2, col. 1 pg. 370, Ng *et al.* discloses a search carried out by calling Bt.Search, by carrying out the steps of acquiring a lock on the node P, on checking that the node is not a root node, acquiring locks for each child and decrementing the counter, releasing the parent lock when the count is zero, and releasing each of the child locks. Thus, Ng *et al.* provides for a parent node to acquire and hold a lock until all the child nodes to be visited acquire locks. Once all the child nodes acquire their locks, the lock on the parent is released. The reference count maintained at the parent decrements every time a lock on the child is acquired. In contrast, the claimed invention allows for a parent lock to be held *until all its child locks are released.* The reference count maintained at the parent decrements *every time a lock on the child is released.* Thus, Ng *et al.* is silent regarding ***as each child lock is released, the reference count decrements by a value of one and the parent lock is released upon release of all child locks associated therewith*** as recited by the subject claims.

In view of the above, it is readily apparent that Joshi and Ng *et al.* alone or in combination, do not teach or suggest all limitations as recited in independent claims 1, 12 and 21 (and the claims that depend from). Accordingly, it is respectfully requested that this rejection should be reversed.

C. Rejection of Claims 2, 3, 10, 15, 18 and 20 Under 35 U.S.C. §103(a)

Claims 2, 3, 10, 15, 18 and 20 stand rejected under 35 U.S.C. §103(a) over Joshi in view of Ng, *et al.* and further in view of Chan, *et al.* (US 6,108,654). Reversal of this rejection is

requested for at least the following reasons. Claims 2, 3, 10, 15 and 18 depend from independent claims 1 and 12. Independent claim 20 recites similar features as independent claim 1. As discussed *supra*, Joshi and Ng *et al.* alone or in combination fail to teach or suggest each and every feature of independent claims 1 and 12. Chan *et al.* relates to finer-grained dynamic allocation and de-allocation of locks in a system, while protecting against abnormal termination that may result in data integrity problems, but fails to cure the deficiencies of Joshi and Ng *et al.* Accordingly, it is requested that this rejection with respect to claims 2, 3, 10, 15 18 and 20 be reversed.

D. Rejection of Claims 6 and 16 Under 35 U.S.C. §103(a)

Claims 6 and 16 stand rejected under 35 U.S.C. §103(a) over Joshi in view of Ng, *et al.* and further in view of Bray, *et al.* Reversal of this rejection is requested for at least the following reasons. Claims 6 and 16 depend from independent claim 1. As discussed *supra*, Joshi and Ng *et al.* alone or in combination fail to teach or suggest each and every feature of independent claim 1. Bray *et al.* relates to systems and a method for locking elements in a hierarchical data structure to allow multiple users at various distributed clients to simultaneously edit unlocked portions of the data structure, but fails to cure the aforementioned deficiencies of ***as each child lock is released, the reference count decrements by a value of one and the parent lock is released upon release of all child locks associated therewith***. Accordingly, it is requested that this rejection with respect to claims 6 and 16 be reversed.

E. Rejection of Claims 26-30 Under 35 U.S.C. §103(a)

Claims 26-30 stand rejected under 35 U.S.C. §103(a) over Joshi in view of Chan, *et al.* (US 6,108,654). Reversal of this rejection is requested for at least the following reasons. Joshi and Chan *et al.* alone or in combination fail to teach or suggest each and every feature of the subject invention.

Appellants' claimed invention relates to systems and methods for supplying a database with a parent-child lock hierarchy arrangement, such that each lock contains sufficient information to determine its own lifetime. In particular, independent claim 26 recites similar features as independent claim 1, namely *the lock manager creates within the parent lock a reference count of the child lock so that* ***the lock manager releases the parent lock upon the***

reference count attainment of a zero value. Joshi and Chan *et al.* are silent regarding such novel features.

As discussed *supra*, with respect to independent claim 1, Joshi fails to disclose the feature of as child locks are released, decreasing the reference counter in the parent by one and determining release of the parent lock based on the child locks.

Chan *et al.* relates to finer-grained dynamic allocation and de-allocation of locks in a system, while protecting against abnormal termination that may result in data integrity problems. At page 13 of the Final Office Action, the Examiner contends that Chan *et al.* teaches such novel features of appellants' claimed invention. Appellants' representative avers to the contrary. At the cited portions, Chan *et al.* discloses recovery domains that allow the lock manager to identify and clean up multiple dubious resource objects as a group. When a process attaches to a recovery domain, a reference count of the domain is incremented. When a dead process detaches, the reference count is decremented. Thus, the reference count is the number of processes currently attached to the domain in the local lock manager instance (See. col. 12, lines 34-38). Thus, Chan *et al.* is silent regarding a reference count maintained by a parent on the number of locked child nodes. In contrast, the claimed invention allows for a lock manager that creates within the parent lock a reference count of the number of child locks, decrements the count as each child lock is released, and releases the parent lock when all the child locks are released. Thus, Chan *et al.* is silent regarding *the lock manager creates within the parent lock a reference count of the child lock* ***so that the lock manager releases the parent lock upon the reference count attainment of a zero value*** as recited by the subject claims.

In view of the above, it is readily apparent that Joshi and Chan *et al.*, alone or in combination, do not teach or suggest all limitations as recited in independent claim 26 (and the claims that depend from). Accordingly, it is respectfully requested that this rejection should be reversed.

F. Conclusion

For at least the above reasons, the claims currently under consideration are believed to be patentable over the cited references. Accordingly, it is respectfully requested that the rejections of claims 1-3, 5-18, 20, 21, 23, 24 and 26-30 be reversed.

If any additional fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063 [MSFTP622US].

Respectfully submitted,
AMIN, TUROCY & CALVIN, LLP

/Himanshu S. Amin/
Himanshu S. Amin
Reg. No. 40,894

AMIN, TUROCY & CALVIN, LLP
57TH Floor, Key Tower
127 Public Square
Cleveland, Ohio 44114
Telephone (216) 696-8730
Facsimile (216) 696-8731

VIII. Claims Appendix (37 C.F.R. §41.37(c)(1)(viii))

1. A computer implemented system for managing the access of system resources in a database comprising the following computer executable components:

a lock manager that acquires a parent lock and one or more child locks on resource(s) of a database, the lock manager stores a reference count of the one or more child locks within the parent lock such that as each child lock is released, the reference count decrements by a value of one, and the parent lock is released upon release of all child locks associated therewith.

2. The system of claim 1, the parent lock is released upon the reference count attainment of a zero value.

3. The system of claim 1, further comprising a lock monitoring system that monitors the reference count of child locks associated with the parent lock.

4. (Cancelled).

5. The system of claim 1 further comprising a lock hierarchy designated by the lock manager.

6. The system of claim 5, the lock hierarchy comprises at least one of a database lock, page lock, table lock and row lock.

7. The system of claim 5 further comprising a page scan optimization that maintains a last child lock until a next one is acquired.

8. The system of claim 1, the parent lock is an intent lock that protects resources at lower levels.

9. The system of claim 5, the one or more child locks is at least one of an exclusive, update and shared lock at a level of the hierarchy.

10. The system of claim 1, the reference count is performed upon completion of at least one of a scan, query or operation.

11. The system of claim 1 further comprising a pointer that can guide a release operation from each child lock to a respective parent lock.

12. A computer implemented method for controlling locks to manage access to system resources in a database comprising:

defining a parent-child relation ship among a plurality of locks in a lock hierarchy;

reference counting one or more child locks associated with a parent lock, such that a parent lock maintains a count reference of respective child locks associated therewith and as each child lock is released, the reference count decrements by a value of one; and,

releasing a parent lock upon a release of all the respective child locks associated therewith.

13. The method of claim 12 the defining act further comprising arranging a top-down lock granularity based on logical or physical granularities of objects stored in the data base.

14. The method of claim 12 further comprising pointing to a parent lock upon releasing a respective child lock associated therewith.

15. The method of claim 12 further comprising reference counting child locks directly associated with the parent lock.

16. The method of claim 12 further comprising acquiring an intent lock at least in one of a table level, page level and database level.

17. The method of claim 12 further comprising maintaining a reference count within a structure of the parent lock.

18. The method of claim 12 further comprising scoping the reference counting of a lock to a transaction.

19. (Cancelled).

20. A computer implemented database management system comprising:

locking means for locking a resource on a database;

means for counting one or more child locks associated with the locking means, wherein the counting means is decreased by one as each child lock is released; and

means for determining a lifetime of the locking means based on the number of child locks associated therewith.

21. A computer implemented method for controlling locks to manage access to system resources in a database comprising:

counting one or more child locks associated with a parent lock to obtain a reference count of the child locks associated therewith;

releasing a child lock;

decrementing the reference count by a value of one; and

releasing the parent lock upon the reference count reaching a zero value.

22. (Cancelled).

23. The method of claim 21 further comprising monitoring the reference count.

24. The method of claim 21 further comprising identifying the parent lock *via* a pointer.

25. (Cancelled).

26. A computer implemented database lock management system for managing access to system resources comprising:

a computer executable lock manager that acquires at least a parent lock and one or more child locks on a database resource, the lock manager creates within the parent lock a reference count of the child lock so that the lock manager releases the parent lock upon the reference count attainment of a zero value.

27. The system of claim 26 further comprising a further computer executable component that monitors the reference count.

28. The system of claim 26 further comprising a forwarding pointer device that identifies a parent lock associated with a released child lock.

29. The system of claim 26 further comprising probabilistic classification models.

30. The system of claim 26, the reference count is the count of direct child locks associated with the parent lock.

IX. Evidence Appendix (37 C.F.R. §41.37(c)(1)(ix))

None.

X. Related Proceedings Appendix (37 C.F.R. §41.37(c)(1)(x))

None.